ISSN No:-2456-2165

# Optimizing the Student Application Process with a Laravel-based System of Waziri Umaru Federal Polytechnic Birnin Kebbi: A Case Study

[*1]Umar Bashar; [*2]Makinde Akindeji Ibrahim (PhD); [3]Adeleye Adedeji & [4]Aminu Dauda
[1]Center for Information Technology, Waziri Umaru Federal Polytechnic, Birnin Kebbi, Nigeria
[2] Department of Information Systems, Federal University of Technology, Akure, Nigeria
[3]Department of Computer Science, Federal University of Technology, Akure, Nigeria
[4]Department of Computer Science Waziri Umaru Federal Polytechnic, Birnin Kebbi, Nigeria

Corresponding Author:- [*1]Umar Bashar; [*2]Makinde Akindeji Ibrahim (PhD)

**Abstract:- The student application process plays a crucial role in the efficient functioning of educational institutions. However, manual and outdated systems often result in delays, errors, and inefficiencies, leading to a poor experience for both applicants and administrators. This case study aims to explore the optimization of the student application process at Waziri Umaru Federal Polytechnic Birnin Kebbi using a Laravel-based system. Application data and users feedback were collected to evaluate the system's performance. Based on the findings, a Laravel-based system was developed and implemented to streamline the student application process. The system offers a user-friendly interface for applicants and also provides an automated workflows, real-time notifications, and centralized data management, enabling efficient application processing and decision-making. The results of the case study indicate significant improvements in the student application process at Waziri Umaru Federal Polytechnic Birnin Kebbi following the implementation of the Laravel-based system. The average increase in applicant satisfaction.**

***Keywords:-*** *Laravel, Optimization, PHP Framework, Student Application Process.*

## I. INTRODUCTION

In recent developments, the integration of technology into educational cycle has shown immense potential for improving the application process and enhancing overall efficiency. The student application process is a critical aspect of any educational institution, as it serves as the gateway for prospective students to enroll and pursue their academic goals (Ghavifekr, and Rosdy, 2015). However, traditional manual methods of handling student applications often lead to inefficiencies, delays, and errors, causing frustration for both applicants and administrative staff.

This paper focuses on optimizing the student application process using a Laravel-based system, with a specific case study conducted at Waziri Umaru Federal Polytechnic Birnin Kebbi. Laravel is a widely adopted PHP framework known for its flexibility, robust features, and simplicity thereby making it an ideal choice for designing efficient and user-friendly web applications. By leveraging the power of Laravel, educational institutions can streamline their application processes, minimize errors, and provide a seamless experience to applicants (Justina et al., 2022)

The research explores the potential benefits of implementing a Laravel-based system, and evaluates the impact of the system on the overall efficiency of the application process. By conducting a case study at Waziri Umaru Federal Polytechnic Birnin Kebbi, this research aims to provide practical insights and development for optimizing student application processes in educational settings.

## II. METHODOLOGY

This section outlines the methodology employed to optimize the student application process using a Laravel-based system. The research methodology incorporates analyzing existing system, design a system architecture for the proposed system, provide design model, and get students feedback on the new system.

The system design phase is a crucial step in optimizing the student application process using a Laravel-based system. It involves translating the identified requirements and challenges into a comprehensive design that addresses the specific needs of the educational institution. The following elements are considered during the system design process:

- User Interface (UI) Design: The UI design focuses on creating a visually appealing and intuitive interface for applicants and administrative staff. It includes designing application forms, document upload sections, notification panels, and dashboard layouts. The UI design is user-friendly, with clear instructions and easy navigation, ensuring a positive user experience.
- Application Form Design: The design of the application form is a critical component of the system. It was structured to capture all necessary applicant information, including personal details, academic history, and

supporting documents. The form design minimizes errors by incorporating validation checks and providing clear instructions for data input.

- Document Upload Functionality: The system allows applicants to upload supporting documents, such as academic transcripts, identification cards, or recommendation letters. The design support various file formats and sizes, provide progress indicators during the upload process, and ensure secure and efficient storage of uploaded documents.
- Workflow Automation: Automating the application workflow helps streamline the process and reduces manual intervention. The system design incorporate features such as automated email notifications to applicants regarding application status updates, acknowledgment emails upon successful submission, and reminders for missing documents. Additionally, the system enables administrators to track and manage applications efficiently.
- Real-time Notifications: Real-time notifications play a vital role in keeping applicants informed about the progress of their applications. The system was designed to send notifications to applicants when their applications are received, processed, or when any action is required from their end. Similarly, administrators receive notifications when new applications are submitted or when specific tasks need attention.
- Centralized Data Management: The system design ensures centralized data management to facilitate easy access and retrieval of application data. This includes designing a secure database schema to store applicant information, providing search and filter options for administrators, and generating reports for data analysis purposes. Data privacy and security measures were also incorporated to protect sensitive information.

➢ *Existing System*

The existing approach employed by the Waziri Umaru Federal Polytechnic Birnin Kebbi lacks an effective communication means between students and other administrator within the institution. The student admission process is tedious and major admission activities are not fully automated. Furthermore, there is limited provision for a medium to disseminate information from the management of the Institution to the students. This situation poses significant challenges in maintaining accurate records, monitoring individual student performance by lecturers, and hampers students' ability to interact with their departments and stay updated on departmental activities.

➢ *New System Analysis*

The system has been segmented into four modules and these include student module, application service module, course module, and database module (As shown in figure 1).

Student Module: The student module is made up of the Login sub-module, registration sub-module, and biodata sub-module.

- Login: This module serves as the entry point to the system. Depending on the type of login, users are granted access to various features and functionalities. The primary purpose is to enable users to utilize the portal.
- Registration: This module allows students to sign up on the application.
- Students: This module constitutes a dedicated webpage for students, enabling them to view, edit, and update their profiles from any location. It also provides platforms for viewing results. It also constitutes a dedicated webpage for students, enabling them to view, edit, and update their profiles from any location. It also provides platforms for viewing results.

Application Service Module: Within this module, the administrator enters their username and password to gain access to the administrator page. The admin serve as the backend for management, registry, student affairs, and other departmental officials, this module enables the management of student records, requests, complaints, and information dissemination to students and other relevant bodies. It also manages the payment record of applicants.

Admission Module: Students can view admission status for the current session through this module, which provides a platform for managing their application. This module provides profile information for various departments associated with the faculties. It offers a comprehensive profile of all the departments, and users can contact these departments through the provided complaint form.

Database Module: Student Addition/Updating/ Deletion: Allows the addition, updating, or deletion of student records based on their respective departments.

ISSN No:-2456-2165

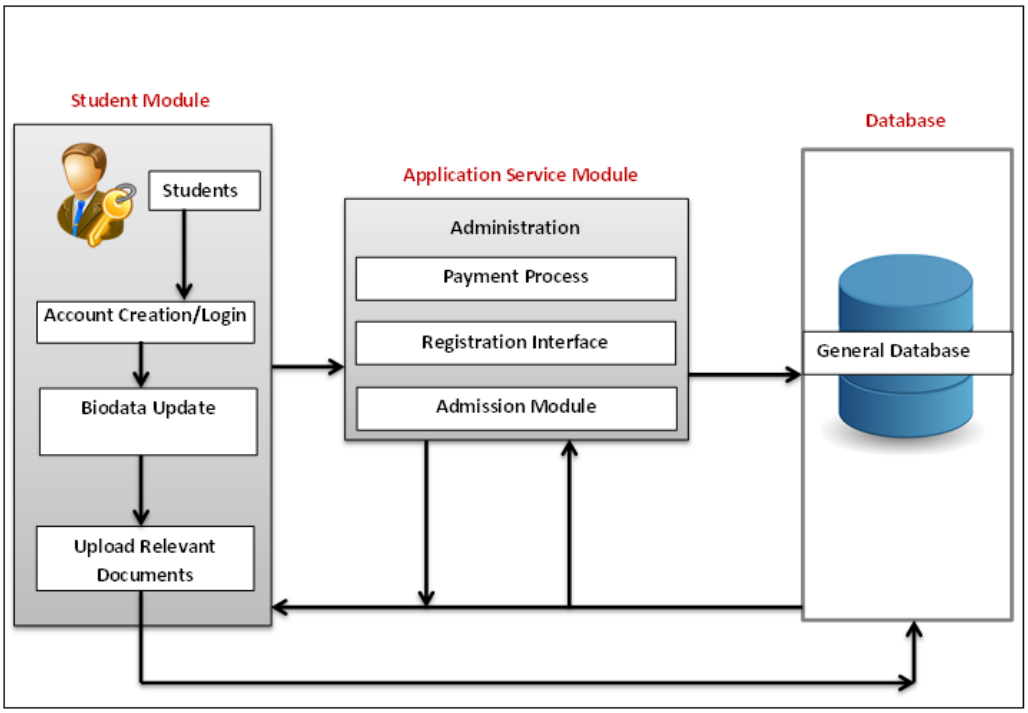


Fig 1 System Architecture of the Application

➢ *System Design Model*

As shown in figure 2, a unified modeling language (UML) was utilized with CASE tools in order to construct a Use Case Diagram model. This approach was taken to enhance the comprehension of the system and establish a cohesive strategy for future development. A use case represents a scenario in which users interact with the system to accomplish a particular task. It delineates the sequence of events involving actors, who are specific types of users, utilizing various aspects of the system's functionality to carry out a process. Each use case encapsulates a specific functional requirement for certain users. By collectively incorporating all the use cases, the overall functional requirements of the system are outlined. Consequently, the initial stage of requirement capture entails representing the requirements in the form of use cases.

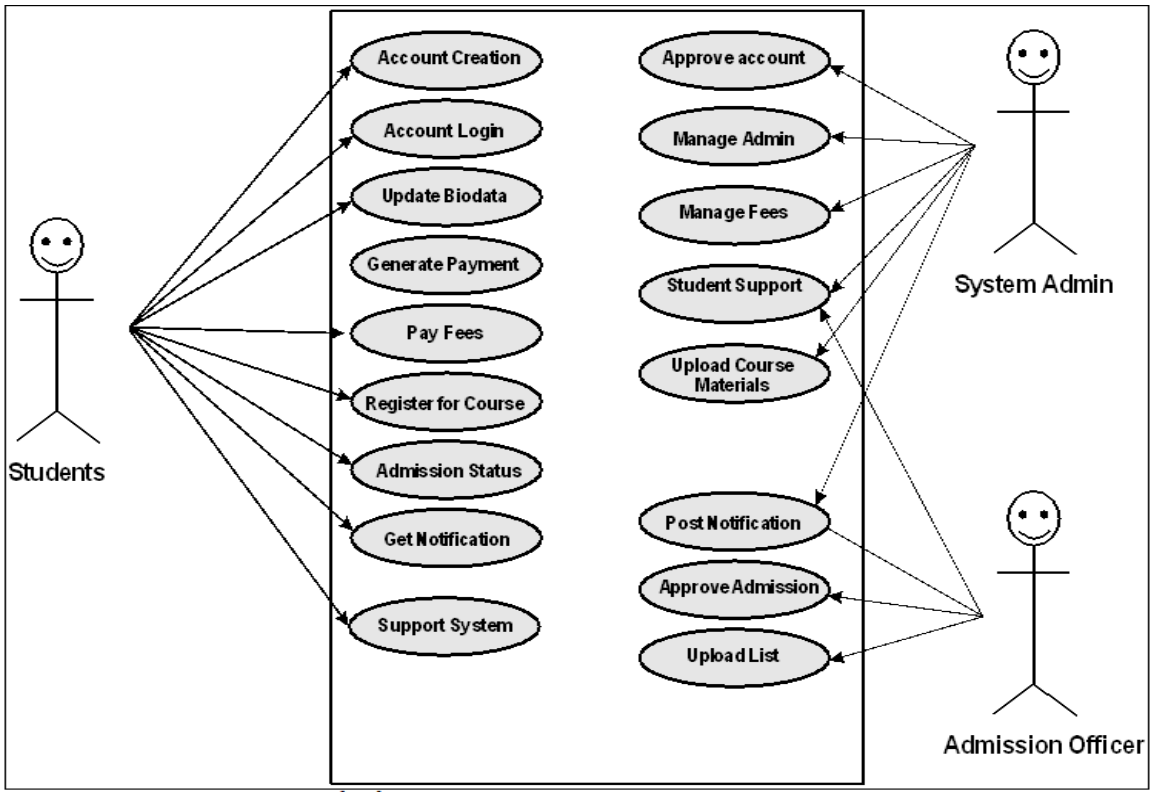


Fig 2 Use Case Diagram for the System

➢ *Implementation*

During the implementation phase, the design concepts are converted into executable code. Developers write computer programs using established programming languages or application generators. Coding is carried out using a range of high-level programming languages such as PHP, MYSQL, SQL, and HTML. The selection of the appropriate programming language is based on the specific requirements and nature of the application being developed. In the case of this project, PHP and JS are utilized as the chosen programming languages.

➢ *System Testing*

During the testing phase, the system undergoes a series of evaluations. Typically, programs are developed as individual modules, each subjected to separate and thorough testing. Subsequently, the entire system is tested as a cohesive unit. The separate modules are integrated and tested as a complete system to ensure the smooth functioning of interfaces between them (integration testing). This phase also involves testing the system on the intended platform and verifying its performance with the anticipated volume of data (volume testing). Furthermore, the system is rigorously tested to ensure it meets the user's requirements and performs as expected.

➢ *Tools and Technologies*

- *Xampp Server*

XAMPP, developed by Apache Friends, is a package that combines open-source web server solutions into a comprehensive stack. It includes the Apache HTTP Server, MariaDB database, and interpreters for PHP and Perl scripting languages. The components used in XAMPP are commonly found in other web server deployments, making the transition from a local test server to a live server easier. XAMPP's simple deployment process allows developers to swiftly install a WAMP or LAMP stack on different operating systems.

- *MySQL*

Structured Query Language (SQL) is a widely recognized abbreviation for a standardized language used to handle and manipulate relational databases. In contrast, MySQL is an open-source Relational Database Management System (RDBMS) that is extensively employed in web applications. It plays a crucial role as part of the popular LAMP (PHP, Linux, Apache, and MySQL) stack, which is a widely adopted open-source software stack for web application development. Unlike Windows, MySQL does not come with a built-in Graphical User Interface (GUI) for administering databases or managing data. Users can either separately download and install MySQL Workbench, a GUI tool, or utilize command line tools.

- *PHP*

PHP is a scripting language employed for the development of various web-based applications, be it dynamic or static websites, or web applications. Initially known as "Personal Home Pages," PHP is now commonly referred to as "Hypertext Pre-processor." PHP scripts are exclusively interpreted on servers equipped with PHP installation. PHP application files are identified by the extension ".php" and contain PHP tags.

- *Php My Admin*

PhpMyAdmin is a freely available open-source tool that is written in PHP, XHTML, CSS, and JavaScript. Its main purpose is to facilitate the administration of MySQL databases through a web interface. PhpMyAdmin offers a range of functionalities, including the ability to create and modify databases, tables, and fields, execute SQL statements, and manage user privileges.

- *The Sublime Text 3 editor*

This is a source code editor that works across multiple platforms and is developed using C++ and Python. Initially, it provides extensive support for various programming and markup languages, and users can extend its functionality via plugins. Symbol indexing allows it to scan files and generate an index, enabling convenient features like "Go to Symbol" and "Go to Definition" to enhance code navigation and exploration.

- *HTML and CSS*

HTML (Hyper Text Markup Language) serves as the fundamental foundation of the World Wide Web, providing structure and meaning to web content. While HTML defines the buildup of a web page, CSS (Cascading Style Sheets) and JavaScript are typically employed to control its appearance/presentation and functionality/behavior, respectively. CSS, specifically, is a stylesheet used to depict the rendering of documents written in HTML or XML, including XML-based formats like MathML, or XHTML. It determines how elements within the document should be rendered on different media, including screens, paper, speech, or other devices. CSS enables precise control over the visual aspects of web pages, allowing developers to define the layout, styling, and visual effects.

- *Bootstrap Framework*

Bootstrap is a highly popular framework utilized for building responsive applications on the web, incorporating HTML, CSS, and JS components. With Bootstrap, websites and applications can be efficiently scaled across various devices, ranging from smartphones and tablets to desktops, thanks to the implementation of CSS media queries.

- *Java Script Framework (Vue)*

Vue is a user interface framework that follows a progressive approach. Unlike traditional monolithic frameworks, Vue is specifically designed to be adopted incrementally. Its core library concentrates solely on the view layer, making it effortless to grasp and integrate with other libraries or ongoing projects. However, Vue is also fully capable of driving advanced Single-Page Applications (SPAs) when utilized alongside modern tools and complementary libraries. This versatility allows developers to leverage Vue for a wide range of projects, from simple integrations to complex and feature-rich applications.

- *Laravel (PHP Framework)*

Laravel, created by Taylor Otwell, is a PHP web framework that operates on an open-source basis. Its purpose is to construct web applications using the model-view-controller (MVC) architectural pattern, with a foundation built upon the Symfony framework. Laravel offers a range of features, including a modular packaging system with a dedicated dependency manager. It provides various methods for interacting with relational databases, as well as utilities that facilitate application deployment and maintenance. Additionally, Laravel is known for its focus on syntactic sugar, which enhances the readability and expressiveness of the code.

## III. LITERATURE REVIEW

The student application process is a fundamental component of educational institutions, and its optimization has garnered significant attention in recent years. With advancements in technology, the integration of web-based systems has become a popular approach to streamline and enhance the efficiency of student application processes. In this literature review, we explore relevant studies that highlight the benefits of optimizing the student application process with a Laravel-based system.

### ➢ *Technology-driven Improvements in Student Application Processes*

Numerous studies have emphasized the advantages of leveraging technology to optimize student application processes. Web-based systems offer a range of features such as online application forms, document uploads, and real-time status tracking, reducing manual paperwork and improving the overall application experience. These systems enhance data accuracy, speed up application processing, and provide a seamless interface for applicants and administrators alike (Griffin et al., 2010).

### ➢ *Role of Laravel in Web Application Development*

Laravel, a popular PHP framework, has gained recognition for its ability to simplify and expedite web application development. It offers a robust set of tools, including database management, routing, and authentication, enabling developers to create efficient and scalable applications. Laravel's modular structure, code reusability, and extensive community support make it an ideal choice for developing web-based student application systems (Purbo, 2021).

### ➢ *MVC Framework*

The architectural pattern known as MVC (Model-View-Controller) is a framework that partitions an application into three distinct sections: the Model, View, and Controller. Each section is designed to handle certain aspects of development. The primary purpose of MVC is to separate the business logic and presentation layer, a concept initially employed in desktop graphical user interfaces (GUIs). However, in contemporary times, MVC architecture has gained popularity in web technology, being widely used for the design of web applications and mobile apps. Several web application frameworks utilize the MVC framework pattern to partition their application's presentation, logic, and data. The MVC framework has long been a favoured choice among developers when designing web or mobile applications.

The MVC design framework emphasizes deep collaboration, wherein classes define or represent only a single type of object, and aims to achieve low coupling, which measures the interrelatedness of classes. In a standard object-oriented programming, mixing presentation and data code is considered a poor practice. Instead, in order to mediate communication between presentation and data classes, controller classes are introduced. These controller classes provide a separation between the presentation layers from the data layer, enabling more specific definitions (Ambler, 2001).

High cohesion offers several advantages, including decreased module intricacy, improved system maintainability and heightened module reusability (Wikipedia, 2016f). By promoting a separation of concerns within the codebase, high cohesion allows for developer specialization, focus, and parallel development when working in teams (Berkeley, 2004).

The MVC typically comprises three classes, as suggested by its name. The controller is the middle link between the view and model classes, coordinating the information flow by receiving input from user to the view while also passing instruction to the model and view to execute actions based on that input. The model is responsible for managing the application's data, commonly involving database operations such as creating, reading, updating, and deleting (CRUD). The responsibility of the view is to display the data derived from the model, often consisting of markup displayed as web pages or other formats like feeds from RSS (Thorpe, 2011). A model acts as an intermediary layer between the data and the application, providing developers with a means to manipulate data. The data itself can be stored in any of database management systems, including MySQL, as well as simpler formats like XML or Excel files. Views serve as the visual depiction of our web application, specifically in the presentation layer. Their primary responsibility is to display the data received by the controller from the model, which contains the business logic. Views can be created effortlessly by utilizing the Blade template language provided by Laravel or by using plain PHP code. Blade operates based on template inheritance and sections, enabling efficient view development.

The main purpose of a controller is to manage requests and transfer data from the model to the views. As a result, the controller serves as the intermediary connecting the model and views. Developers have the flexibility to choose whether to implement their business logic in routers or controllers. Routers can be advantageous for smaller web applications or when quickly defining static pages. It is not essential to create controllers for every individual page of the web application.

Laravel is utilized for designing and enhancing systems developed, thereby reducing time and costs. It offers distinctive features that empower developers to create web-based systems while preserving data integrity and functionality (Armel, 2014). With Laravel, developers can build simple yet well-designed processes or systems, leveraging its advanced and elegant features as a PHP framework (Ibrahim et al., 2018). Among its numerous benefits, Laravel has gained popularity as one of the leading PHP frameworks, enables the creation of robust layouts, and facilitates the development of secure applications. As a comprehensive MVC framework, Laravel has swiftly become the industry standard, widely adopted in modern development environments.

- *Web Application Frameworks*

In the field of web application development, a framework is a collection of libraries and established practices aimed at easing repetitive and common tasks. Frameworks are designed to enhance security, expedite development speed, and facilitate the creation of modularized code bases for easier maintenance. While different frameworks may employ varying approaches, they generally align with this fundamental principle (Saeed, and Haidar, 2019).

Frameworks commonly offer several features to streamline development processes. They provide a simplified and platform-independent approach to connect with databases, interact with them, and validate database actions. Additionally, frameworks offer techniques to manage session data, authenticate users, and safeguard authenticated sessions. Implementing page caching to alleviate server load, mapping or re-routing URLs for cleaner and user-friendly URLs, employing a templating system to minimize client-side markup, and generating automated scaffolding for common database-backed code structures are among the typical functionalities provided by frameworks (Giffin, 2017).

- *Case Studies on Laravel-based Student Application Systems*

Several case studies have highlighted the successful implementation of Laravel-based systems in optimizing student application processes (Ashiraf, 2017). These studies demonstrate the effectiveness of Laravel in improving efficiency, reducing processing times, and enhancing applicant satisfaction. Laravel's features, such as automated workflows, centralized data management, and user-friendly interfaces, contribute to a streamlined and error-free application process.

- *User Experience and Applicant Satisfaction*

A key focus of optimizing student application processes is improving the user experience and applicant satisfaction. Studies have shown that web-based systems, including those built with Laravel, significantly enhance the overall experience for applicants. User-friendly interfaces, clear instructions, and real-time notifications contribute to higher applicant satisfaction, leading to increased enrollment rates and positive institutional reputation ( Pandita and Kiran, 2023).

- *Data Security and Privacy*

The need for security and privacy of applicant data is crucial in student application processes. Laravel provides robust security features, including encryption, cross-site scripting prevention, and SQL injection protection, ensuring the integrity and confidentiality of sensitive information (Adamu et al., 2020). Studies emphasize the importance of implementing strong security measures to protect applicant data and maintain compliance with privacy regulations.

The integration of technology, specifically Laravel's features and capabilities, enhances efficiency, reduces processing times, improves user experience, and ensures data security (Adamu et al., 2020). The studies highlight the positive impact of Laravel-based systems on applicant satisfaction, institutional reputation, and overall operational effectiveness. Implementing a Laravel-based system for student applications can serve as a valuable solution for educational institutions seeking to streamline their processes and provide an enhanced experience for applicants and administrative staff.

## IV. DISCUSSIONS AND RESULTS

The implementation of a Laravel-based system for optimizing the student application process at Waziri Umaru Federal Polytechnic Birnin Kebbi resulted in significant improvements in efficiency, user experience, and overall effectiveness. The Laravel-based system streamlined the application process by providing a user-friendly interface for applicants. The online application form, intuitive navigation, and clear instructions reduced errors and improved the overall application experience. Applicants found it easier to complete the application process, resulting in increased applicant satisfaction.

The system's workflow automation features reduced manual intervention and administrative tasks. Automated email notifications were sent to applicants at different stages of the application process, providing real-time updates on the status of their applications. This automation improved communication, reduced processing times, and increased transparency for both applicants and administrators.

The system's centralized data management capabilities facilitated easy access and retrieval of application data for administrators. With a secure database structure, search and filter options, and reporting functionalities, administrators could manage and analyze application data more effectively. This centralized data management enhanced decision-making processes and improved data accuracy.

The Laravel-based system implemented robust security measures to protect applicant data. Features such as encryption, SQL injection prevention, and cross-site scripting prevention ensured the confidentiality and integrity of sensitive information. This enhanced data security helped build trust among applicants and complied with privacy regulations.

ISSN No:-2456-2165

The application is developed using the Laravel framework of PHP. The development server can be started by running the php artisan serve command in the terminal.

➢ *Registration Interface*

To register (Figure 3), the user must provide their first name, last name, middle name (if applicable), phone number, email, and the program they wish to apply for. Currently, there are four programs available:

- Higher National Diploma (HND)
- National Diploma (ND)
- National Certificate of Education (NCE)
- Polytechnic Diploma(PD)

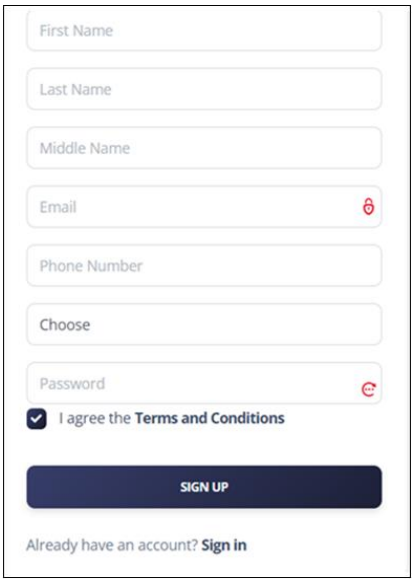


Fig 3 Registration Interface

Upon successful registration, the user is redirected to the login page. But for unsuccessful registration, an error message will be displayed in red with information on what went wrong.

➢ *Login Interface*

To login (Figure 4), a student must provide their registered phone number and password. The reason for the use of phone number is for students to easily remember it.

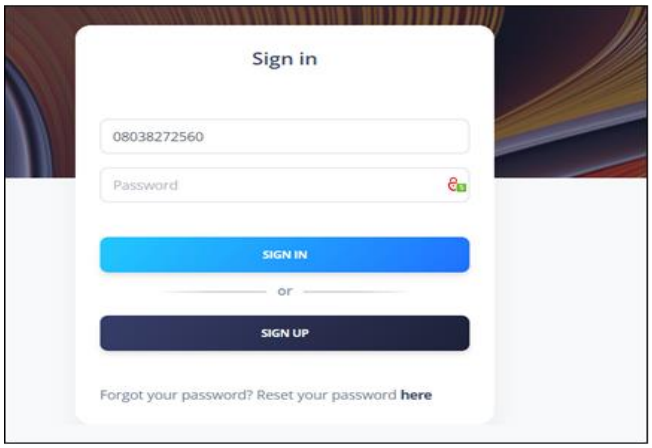


Fig 4 Login Interface

Upon successful login, the user is redirected to a series of steps to fill in all the required details for admission. This includes personal details, address, and academic history. This part is categorized into five stages. The stages vary according to the program a student chooses. For example, a student is only required to fill in details on higher school certificate if they choose Diploma or Higher National Diploma, respectively. Some of the steps are captured as shown in Figure 5:

- *Personal Details*
- *Address*
- *Academic History*

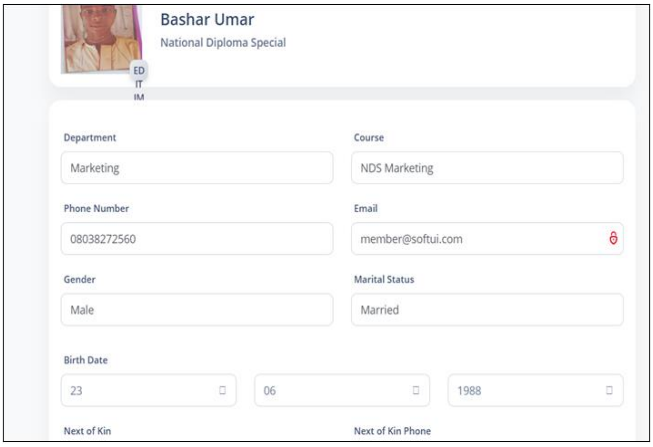


Fig 5 Personal Details Capturing Interface

After completing all personal details steps, the user can update the information later provided the admission process has not started. After filling in all the required details, the user is redirected to a dashboard (This can be viewed in Figure 6). The dashboard contains several features, including:

- Department and course selected
- Program chosen
- Invoices or payments made (where applicable)
- Admission status

The user can navigate to any part of the portal through the dashboard.

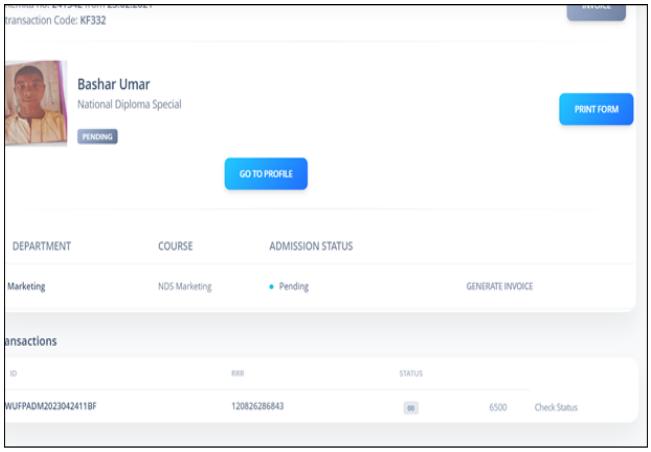


Fig 6 User Dashboard

ISSN No:-2456-2165

➢ *Payment Invoice*

From the dashboard, students have the option to generate an invoice for payment of admission fees. The admission fee varies depending on the program chosen by the student. For the National Diploma (ND) and National Certificate of Education (N.C.E) programs, students are only required to pay a screening fee of N2500. However, for other programs such as the Higher National Diploma (HND), National Diploma Special (NDS), and Polytechnic Diploma, the fee amount is N5600 and N6500 respectively.

To generate an invoice, you can click on the "Generate Invoice" button, which is highlighted in yellow on the dashboard as shown in figure 7.


Fig 7 Generate Inovice

Clicking the button will take you to the invoice page (figure 8), where you can review all the payment details, including the transaction ID, name, amount, Remita number, and more.


Fig 8 Sample Invoice Page

Once you have verified that all the details are correct, you can proceed to make the payment through the Remita gateway API (Figure 9).

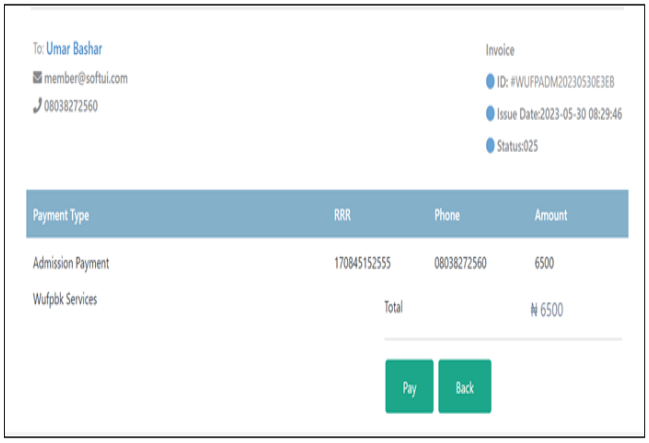


Fig 9 Payment Gateway

Upon successful payment, a receipt will be generated, and the invoice details will be stored in the database. You will then be redirected back to the main dashboard, where you can find a button to print your admission form. Please note that this button will only appear if the payment has been successfully made.

In order to assess the system's impact on the ease of registration and other online activities, the research study utilized a survey to collect feedback from students. The survey consisted of 8 questions that aimed to gauge students' experiences with the system. These questions can be categorized into two main groups, with 5 questions focusing on students' personal records and another 5 addressing the communication between students and the application. The second category includes the following specific questions:

- Did the application improve communication?
- Did the application enhance the role of the system administrator?
- Did the application provide appropriate feedback?
- Did the application offer proper functionality for uploading?
- Did the application contribute to workflow improvement?

Table 1 and Figure 10 indicate that the entire group of 65 participants expressed their opinion on the application's impact on system administration, with no participants holding a negative stance. Specifically, the data reveals that 7% of respondents felt neutral, whereas 21% agreed and 72% strongly agreed that the application has enhanced system administration.

ISSN No:-2456-2165

Table 1 Survey Questions Frequencies

| Survey Question | Strongly Disagree | Disagree | Neutral | Agree | Strongly Agree |
|---|---|---|---|---|---|
| Application improved communication | 0 | 0 | 0 | 26% | 74% |
| Application improved system administration | 0 | 0 | 7% | 21% | 72% |
| Proper Feedback | 0 | 0 | 5% | 31% | 64% |
| Proper Upload Functionality | 0 | 0 | 9% | 31% | 60% |
| Workflow Improvement | 0 | 0 | 7% | 26% | 67% |

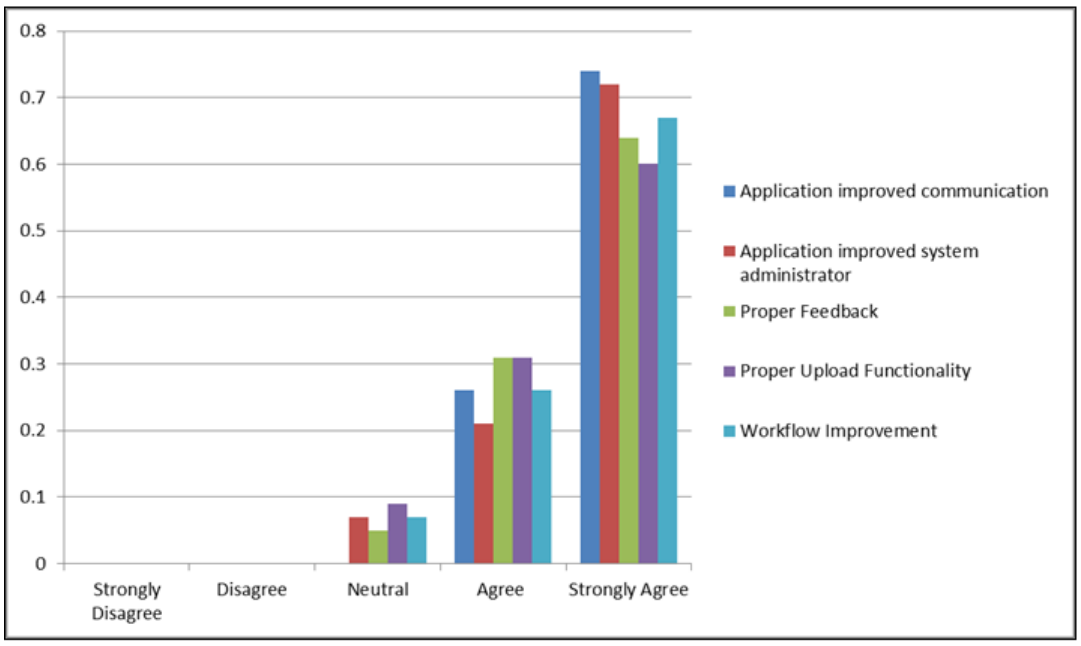


Fig 10 Survey on Improved Learning Performance

When evaluating the application's effect on communication, all survey participants, totaling 100%, reported agreement or strong agreement. Among them, 26% agreed and 74% strongly agreed that the application had improved communication. Importantly, no participants disagreed with the notion that the application improved communication (See Table 2).

Regarding the application's feedback, a significant majority of 95% agreed or strongly agreed with its improvement, while only 5% maintained a neutral viewpoint when asked about this aspect.

Table 2 Descriptive Statistics of the Survey Questions

| Survey Question | N | Mean | Median | Standard Deviation |
|---|---|---|---|---|
| Application improved communication | 65 | 4.74 | 5 | 0.45 |
| Application improved system administration | 65 | 4.64 | 5 | 0.65 |
| Proper Feedback | 65 | 4.60 | 5 | 0.59 |
| Proper Upload Functionality | 65 | 4.50 | 5 | 0.67 |
| Workflow Improvement | 65 | 4.60 | 5 | 0.63 |

## V. CONCLUSION

The study's findings have broader implications for other educational institutions seeking to optimize their student application processes. By leveraging Laravel's features and best practices, institutions can streamline their processes, improve data management, and enhance the overall experience for applicants and administrative staff. Further research can explore the long-term impact of the implemented system and assess its scalability and adaptability in different educational settings.

## ACKNOWLEDGEMENT

I wish to thank the management of Waziri Umaru federal polytechnic for their input and support. I will also like to extend my gratitude to all my co-authors for their sincere advice and contribution.

- *Funding*

The author(s) did not receive any funding during the research.

- *Author Contribution*
- *Conflict Interest*

Author(s) did not have any conflict of interest.

## REFERENCES

[1]. Adamu, J., Hamzah, R., & Rosli, M. M. (2020). Security issues and framework of electronic medical record: A review. Bulletin of Electrical Engineering and Informatics, 9(2), 565-572. DOI: https://doi.org/10.11591/eei.v9i2.2064

[2]. Ambler, S. 2001. "The Object Primer." 170-172. Cambridge University Press. DOI: https://doi.org/10.1017/CBO9780511584077

[3]. Armel, J. (2014). Web application development with Laravel PHP Framework version 4. Media Engineering, (April), 59. Retrieved from http://urn.fi/URN:NBN:fi:amk-201405066158

[4]. Ashiraf, S. S. (2017). A Resource Management System For Bishop Cipriano Kihangire Secondary School Library (doctoral dissertation, makerere university). https://asbatlibrary.s3.eu-central-1.amazonaws.com/49bb3dd5-facc-482c-abe1-fdd74ecbbc88-Ssemugabi.%20S.Ashiraf-A%20Resource%20Management%20System%20for%20Bishop%20Cipriano%20Kihangire%20Secondary%20School%20Library.pdf

[5]. Berkeley, UC. (2004). Model-View-Controller: A Design Pattern for Software. 06. https://ist.berkeley.edu/as-ag/pub/pdf/mvc-seminar.pdf.

[6]. Ghavifekr, S., & Rosdy, W. A. W. (2015). Teaching and learning with technology: Effectiveness of ICT integration in schools. International journal of research in education and science, 1(2), 175-191. DOI:10.21890/ijres.23596

[7]. Giffin, D., Levy, A., Stefan, D., Terei, D., Mazieres, D., Mitchell, J., & Russo, A. (2017). Hails: Protecting data privacy in untrusted web applications. Journal of Computer Security, 25(4-5), 427-461. https://doi.org/10.3233/JCS-15801

[8]. Griffin, G., Kim, Y. E., & Turnbull, D. (2010). Beat-sync-mash-coder: A web application for real-time creation of beat-synchronous music mashups. In 2010 IEEE International Conference on Acoustics, Speech and Signal Processing (pp. 437-440). IEEE. DOI: 10.1109/ICASSP.2010.5495743

[9]. Ibrahim, A. F., Musa, N., & Jamaludin, M. N. F. (2018). Internship application system (IAS) for university students using laravel. Journal of Computing Research and Innovation (JCRINN), 3(3), 12-18. DOI:10.24191/jcrinn.v3i4.85

[10]. Justina, I. A., Oyekan, E. A., & Orogbemi, O. M. (2022). A Secured Cloud-Based Electronic Document Management System. International Journal of Innovative Research and Development, 11(12). https://doi.org/10.24940/ijird/2022/v11/i12/DEC220100

[11]. Pandita, A., & Kiran, R. (2023). The Technology Interface and Student Engagement Are Significant Stimuli in Sustainable Student Satisfaction. Sustainability, 15(10), 7923. https://doi.org/10.3390/su15107923

[12]. Purbo, O. W. (2021). A Systematic Analysis: Website Development using Codeigniter and Laravel Framework. Enrichment: Journal of Management, 12(1), 1008-1014. https://doi.org/10.35335/enrichment.v12i1.346

[13]. Saeed, H. M., & Haidar, H. M. (2019). Developing an integrated admission system for universities. Journal of Engineering and Applied Sciences, 14(15), 5329-5335. DOI: 10.18502/kss.v3i12.4076

[14]. Thorpe, S. 2011. Untangling MVC with CodeIgniter. Accessed 6 18, 2023. https://www.sitepoint.com/untangling-mvc-with-codeigniter/.

[15]. Wikipedia. 2016a. Cohesion (computer_science). Accessed 6 18, 2023. https://en.wikipedia.org/wiki/Cohesion_(computer_science).